An Optimized Initialization Algorithm to Ensure Accuracy in Quantum Monte Carlo Calculations

DANIEL R. FISHER, DAVID R. KENT IV, MICHAEL T. FELDMANN, WILLIAM A. GODDARD III

Materials and Process Simulation Center, Beckman Institute (139-74), Division of Chemistry and Chemical Engineering, California Institute of Technology, Pasadena, California 91125

Abstract: Quantum Monte Carlo (QMC) calculations require the generation of random electronic configurations with respect to a desired probability density, usually the square of the magnitude of the wavefunction. In most cases, the Metropolis algorithm is used to generate a sequence of configurations in a Markov chain. This method has an inherent equilibration phase, during which the configurations are not representative of the desired density and must be discarded. If statistics are gathered before the walkers have equilibrated, contamination by nonequilibrated configurations can greatly reduce the accuracy of the results. Because separate Markov chains must be equilibrated for the walkers on each processor, the use of a long equilibration phase has a profoundly detrimental effect on the efficiency of large parallel calculations. The stratified atomic walker initialization (STRAW) shortens the equilibration phase of QMC calculations by generating statistically independent electronic configurations in regions of high probability density. This ensures the accuracy of calculations by avoiding contamination by nonequilibrated configurations. Shortening the length of the equilibration phase also results in significant improvements in the efficiency of parallel calculations, which reduces the total computational run time. For example, using STRAW rather than a standard initialization method in 512 processor calculations reduces the amount of time needed to calculate the energy expectation value of a trial function for a molecule of the energetic material RDX to within 0.01 au by 33%.

© 2008 Wiley Periodicals, Inc.   J Comput Chem 29: 2335–2343, 2008

Key words: quantum Monte Carlo; walker initialization; parallel computing; parallel efficiency

Introduction

Quantum Monte Carlo (QMC) methods for simulating the electronic structure of molecules[1,2] can in principle provide energies to within chemical accuracy ($\approx$ 2 kcal/mol).[3–5] The computational expense of QMC scales with system size as $O(N^3)$ or better,[6–9] albeit with a large prefactor. This is much more favorable than other quantum mechanical methods capable of comparable accuracy, such as coupled cluster, which tend to scale very poorly with the size of the system, generally $O(N^6$ to $N!)$.[10] Moreover, the stochastic nature of QMC makes it relatively easy to parallelize over a large number of processors, which can allow calculations to finish in a reasonable amount of time despite the slow convergence of Monte Carlo.

As supercomputing resources improve and become more accessible to researchers,[11,12] QMC will become a powerful tool for conducting accurate simulations on chemically interesting systems. Recent efforts have focused making these calculations more straightforward and efficient on heterogeneous and homogeneous computers. To this end, a finite all-electron QMC program, QMcBeaver, has been written and used to develop and demonstrate several new algorithms.[13–15]

Before statistics gathering begins in a QMC calculation, the walkers must be allowed to equilibrate so that their configurations are proportional to the desired density. It is impossible to calculate

accurate expectation values if nonequilibrated configurations contaminate the statistics. To ensure their statistical independence, the walkers must equilibrate separately. This makes the equilibration phase a serial step of the calculation and a major limiting factor in the efficiency of parallel calculations. These considerations make it imperative that the equilibration process be fast and reliable. For example, we show that for the energetic material RDX, approximately 30,000 iterations are necessary for equilibration when the initial configurations are generated by a standard method.

We present here a simple method for choosing initial electronic configurations designed to reduce the length of the equilibration phase of calculations. The stratified atomic walker initialization (STRAW) for QMC calculations uses a shell model to distribute the electrons. When STRAW is used in RDX calculations, 100 iterations are sufficient for equilibration.

Avoiding contamination by nonequilibrated configurations in QMC calculations improves their accuracy, and reducing the cost of equilibration makes calculations with large numbers of processors much more efficient. Improving the parallel efficiency of these calculations makes better use of computer resources and will broaden the range of systems for which QMC calculations are practical.

The Metropolis Algorithm and the Initialization Catastrophe

QMC calculations center around the random generation of electronic configurations with respect to quantum mechanical probability densities. In this work, we focus on variational Monte Carlo (VMC), in which the trial wavefunction is sampled in order to optimize its adjustable parameters.[16–18]

VMC trial functions usually have the form $\Psi_{\text{VMC}} = \Psi_{\text{SCF}}J$, where Ψ_{SCF} is one or a sum of Slater determinant wavefunctions obtained by a standard electronic structure method such as Hartree-Fock (HF), density functional theory (DFT), or multiconfiguration self consistent field (MSCSF). The Jastrow factor, J,[19–21] is a symmetric function of the interparticle coordinates meant to account for quantum mechanical cusp conditions[22] and short range correlations.

The expectation value for the energy of this trial function is

$$\langle E \rangle = \frac{\langle \Psi_{\text{VMC}} | \hat{H} | \Psi_{\text{VMC}} \rangle}{\langle \Psi_{\text{VMC}} | \Psi_{\text{VMC}} \rangle} = \frac{\int_{-\infty}^{\infty} d\vec{x}\, \Psi_{\text{VMC}}^*(\vec{x}) \hat{H} \Psi_{\text{VMC}}(\vec{x})}{\int_{-\infty}^{\infty} d\vec{x}\, \Psi_{\text{VMC}}^*(\vec{x}) \Psi_{\text{VMC}}(\vec{x})}, \quad (1)$$

where $\vec{x}$ is a $3N$ dimensional vector of the positions of the N electrons in the molecule. Because the Jastrow factor includes explicit interparticle coordinates, eq. (1) cannot be separated into independent electron problems and solved using the standard SCF procedure. Instead, the expectation value must be evaluated using a stochastic method.[23]

The local energy of a configuration, $\vec{x}$, is defined as $E_{\text{L}}(\vec{x}) = \frac{\hat{H}\Psi_{\text{VMC}}(\vec{x})}{\Psi_{\text{VMC}}(\vec{x})}$. Using this quantity, the expectation value of the energy can be rewritten:

$$\langle E \rangle = \frac{\int_{-\infty}^{\infty} d\vec{x}\, |\Psi_{\text{VMC}}(\vec{x})|^2 E_{\text{L}}(\vec{x})}{\int_{-\infty}^{\infty} d\vec{x}\, |\Psi_{\text{VMC}}(\vec{x})|^2} = \int_{-\infty}^{\infty} d\vec{x}\, \rho_{\text{VMC}}(\vec{x}) E_{\text{L}}(\vec{x}) \quad (2)$$

with

$$\rho_{\text{VMC}}(\vec{x}) = \frac{|\Psi_{\text{VMC}}(\vec{x})|^2}{\int_{-\infty}^{\infty} d\vec{x}\, |\Psi_{\text{VMC}}(\vec{x})|^2} \quad (3)$$

The expectation value now has the form of a weighted average. A series of M electronic configurations, $\{\vec{x}_i\}$, is generated with respect to ρ_{VMC} and used to evaluate the expectation value of the energy:

$$\langle E \rangle = \frac{1}{M} \sum_{i=1}^{M} E_{\text{L}}(\vec{x}_i) \pm O\left(\frac{1}{\sqrt{M}}\right) \quad (4)$$

ρ_{VMC} is an extremely complicated, $3N$-dimensional probability density. An effective way to generate electronic configurations with respect to this type of function is to use a Markov chain, which is defined in terms of the transition probability $T(\vec{x} \to \vec{y})$ of having the configuration $\vec{y}$ after $\vec{x}$ in the chain. The Metropolis algorithm[24] is a method for generating a Markov chain of points distributed with respect to a desired probability density. It states that a Markov chain will converge to a desired density, $f(\vec{x})$, if its transition probabilities satisfy the following relationship:

$$T(\vec{x} \to \vec{y})f(\vec{x}) = T(\vec{y} \to \vec{x})f(\vec{y}) \quad (5)$$

Equation (5) is known as the detailed balance condition. The most commonly used formula for calculating the probability of accepting a proposed move from $\vec{x}$ to $\vec{y}$ that satisfies detailed balance is

$$A(\vec{x} \to \vec{y}) = \min\left[1, \frac{w(\vec{y} \to \vec{x})f(\vec{y})}{w(\vec{x} \to \vec{y})f(\vec{x})}\right], \quad (6)$$

where $w(\vec{x} \to \vec{y})$ is the probability for proposing a move from $\vec{x}$ to $\vec{y}$.

In this work, we use the accelerated Metropolis algorithm developed by Umrigar et al.[25, 26] to propose configurations and calculate $w(\vec{x} \to \vec{y})$. This algorithm allows different length scales for the motions of core and valence electrons, which increases the size of the time step that can be used in a calculation while maintaining a high acceptance rate.

The Metropolis algorithm guarantees that the Markov chain will equilibrate to the desired distribution, but does not provide any criteria to predict the number of iterations that will be necessary for equilibration or to determine when it has occurred. It is vital to avoid contamination by nonequilibrated points in calculations, because it is impossible to calculate accurate expectation values using configurations that do not represent the desired density.

The equilibration time will depend strongly on the choice of the initial configuration, $\vec{x}_0$. If $\vec{x}_0$ is in a region of low probability density, repeated iterations using eq. (6) will guide the chain into regions of higher probability density. The chain is equilibrated when it reaches a region whose probability density is high enough that sampling it is consistent with the desired probability density and the total number of iterations. Clearly, the number of iterations required for

equilibration can be minimized by making an intelligent choice for $\vec{x}_0$.

In Monte Carlo simulations, a *walker* is an entity that defines the state of the system at a particular instant. In QMC, a walker consists of a $3N$-dimensional electronic configuration. An ensemble of walkers is used to carry out the integration, with each one tracing out an independent Markov chain in configuration space. In a parallel calculation, an ensemble of walkers is equilibrated and propagated on each processor, and the results are gathered to obtain the global results.

In the QMcBeaver program, each processor must have at least one walker, and the number of walkers per processor is a user defined constant. Since the number of walkers increases linearly with the number of processors, the computational effort devoted to equilibration increases as well. The impact of the equilibration phase on the efficiency of a parallel calculation was predicted and demonstrated by Feldmann and Kent,[15] and we follow their derivation.

Since separate Markov chains must be equilibrated on each processor, the total equilibration time scales as $O(N_{\text{Processors}})$. The time devoted to generating statistics, $T^{\text{Propagate}}$, scales as $O(1)$ because the number of independent samples needed to achieve a certain level of convergence does not change with the number of processors. From this, the efficiency, or fraction of the total calculation time devoted to useful work, ϵ, is

$$\epsilon = \frac{T^{\text{Propagate}}}{T^{\text{Initialize}} + T^{\text{Equilibrate}} + T^{\text{Synchronize}} + T^{\text{Communicate}} + T^{\text{Propagate}}} \tag{7}$$

$$\approx \frac{O(1)}{O(N_{\text{Processors}}) + O(1)} \tag{8}$$

Since the synchronization and communication costs for QMC calculations are extremely small, the main threat to efficiency in parallel calculations will be the equilibration time. To use a large number of processors efficiently, an algorithm for quickly generating equilibrated, statistically independent electronic configurations for the walkers is necessary. The next section examines how initial walkers are generated in several QMC programs and considers possibilities for improvement.

Walker Initialization

The walker initialization algorithm originally implemented in the QMcBeaver program works as follows: the electrons of the molecule are assigned to the nuclei according to the density implied by the SCF wavefunction. Each nucleus and its electrons are treated as an atom, and the electrons are distributed with respect to a three-dimensional Gaussian centered on the nucleus whose variance is related to the covalent radius of that atom. The configuration is discarded and a new one is generated if substituting the locations of the electrons into the Slater determinant part of the wavefunction results in a singularity.[13] This happens if there is any linear dependence among the columns of the determinant, which can happen if two parallel spin electrons are too close to each other. We will refer to this method as the Gaussian Atomic Walker Initialization (GAWI).

The initialization algorithm of Casino, a QMC program developed at Cambridge, assigns the electrons to atoms and then places the electrons randomly within spheres centered on the atoms.[27]

QMAGIC, a QMC program developed at UC Berkeley and the Lawrence Berkeley National Laboratory, uses an initialization method similar to GAWI.[28] The electrons are distributed with respect to three-dimensional Gaussians centered on the nuclei, and then the configuration is checked to ensure no two particles are closer than a tolerance distance to each other. Zori, a new QMC program developed in the same research group, distributes electrons randomly in spheres of the atomic covalent radius and checks to make sure no electron–electron distance is smaller than a threshold. A configuration is discarded if its local energy is not within a given range of an estimate of the energy of the system.[29] This *ad hoc* test is probably effective in eliminating some unfavorable initial configurations, but requires additional user specified parameters and could cause the walker initialization to scale badly if a large fraction of the configurations generated were discarded.

These initialization methods give satisfactory performance in calculations on small molecules using moderately large computers. In these calculations, the equilibration phase is a small part of the total computational expense and does not have a severely detrimental effect on the efficiency. As the size of the molecules and the number of processors increase, however, the fraction of the total time spent equilibrating can become significant. By improving the way initial configurations are chosen, the length of the equilibration phase can be reduced, which will improve the parallel scaling and efficiency of calculations using large numbers of processors.

The walker initialization algorithms described earlier suffer from several deficiencies. Most importantly, because all the electrons of an atom are distributed with respect to the same probability distribution, the electrons tend not to avoid each other in the initial configurations. For opposite spin electrons, this is unfavorable because of their coulomb repulsion. For parallel spin electrons, however, it is even worse. The antisymmetry of the wavefunction dictated by the Pauli principle forces the wavefunction to go to zero as two parallel spin electrons approach each other. In addition, these methods ignore the structure of the energy levels, in which there will be certain numbers of electrons mostly within annular shells.

Because the initialization methods of this section share these deficiencies, we assume that their performance will be similar, and will use GAWI to represent them in comparisons.

STRAW

The Stratified Atomic Walker initialization (STRAW) is a method for generating initial electronic configurations that addresses the problems described above. In STRAW, the electrons are assigned to the nuclei as in the other methods. Care is taken to ensure that, for an overall neutral molecule, each atom is neutral. The atoms are treated separately, and the electrons are partitioned into energy levels, with one alpha spin and one beta spin electron in the first energy level, up to four alpha spins and four beta spins in the second energy level, and so on. The electrons in each energy level are distributed using the transformation method with respect to probability densities in spherical coordinates: r, θ, ϕ. The transformation method directly converts uniform random numbers on the interval $(0, 1)$ to random

numbers distributed with respect to a desired probability density using the inverse of its cumulative distribution function.[30]

To obtain the radial densities for the energy levels, Hartree-Fock/6-311G** calculations were carried out for each atom in the first three rows of the periodic table using Jaguar.[31] The occupied atomic orbitals were localized by the Boys procedure, which creates orbitals with maximum insensitivity to changes in distant nuclear charges.[32] For molecules, the resulting orbitals are localized around the chemical bonds and in the atomic lone pair regions. In our case, the Boys procedure hybridizes the valence orbitals of the atom.

A representative orbital for each energy level was chosen and expressed as a sum of primitive Gaussians:

$$\psi = \sum_i d_i x^{a_i} y^{b_i} z^{c_i} \exp(-\alpha_i r^2) \tag{9}$$

The d_i are the expansion coefficients and the exponents a_i, b_i, and c_i determine the symmetry of the primitive Gaussians. The square of the orbital is its probability density:

$$\rho = |\psi|^2 = \sum_{i,j} d_i d_j x^{a_i+a_j} y^{b_i+b_j} z^{c_i+c_j} \exp[-(\alpha_i + \alpha_j)r^2] \tag{10}$$

Converting the probability density into spherical coordinates and integrating over the angles yields the radial marginal probability distribution of the orbital:

$$R(s) = \sum_{i,j} d_i d_j \int_0^{2\pi} d\phi \cos\phi^{a_i+a_j} \sin\phi^{b_i+b_j}$$
$$\times \int_0^{\pi} d\theta \sin\theta^{a_i+a_j+b_i+b_j+1} \cos\theta^{c_i+c_j}$$
$$\times \int_0^s dr\, r^{2+a_i+a_j+b_i+b_j+c_i+c_j} \exp[-(\alpha_i + \alpha_j)r^2] \tag{11}$$

The integrals over the angles were done analytically, and the radial integrals were evaluated numerically by a change of variables from the incomplete gamma function. Because the orbitals are normalized and their probability densities are always positive, $R(s)$ increases monotonically with r from zero to one.

Radial probability distributions for each energy level of each atom were tabulated and inverted by interchanging the coordinates. For example, the inverted radial marginal distributions for the first and second energy levels of carbon are shown in Fig. 1. To generate the radial distances for the electrons in an energy level, QMcBeaver fits a cubic spline to the appropriate tabulated inverted distribution. A uniform number in the range (0, 1) is generated for each electron and converted to a radial coordinate by evaluating the spline.

The transformation method is also used to generate the angular coordinates for the electrons. Probability densities in θ and ϕ for s, sp, sp^2, and sp^3 hybrid orbitals were found in terms of the real spherical harmonics[33] and integrated analytically. The results were tabulated and inverted. As with the radial distributions, splines are fit to the tabulated inverted distributions and used to generate the angular coordinates of the electrons in the energy level.
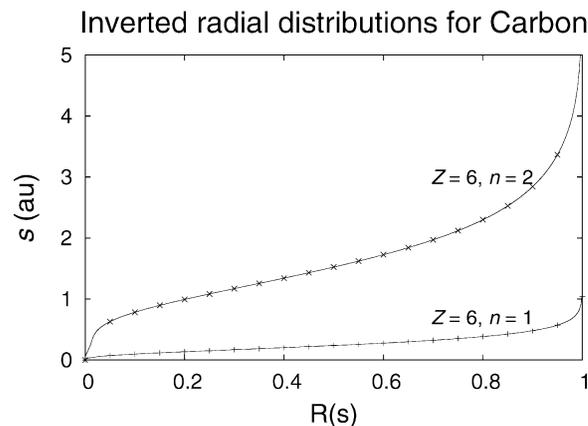
Figure 1. The inverted radial distributions for the first and second energy levels of carbon. To generate the radial coordinate for an electron in one of these energy levels, we generate a uniform random number in the range (0,1) and then evaluate the appropriate inverted distribution.

The probability densities in θ and ϕ are chosen for each electron so that they avoid each other, with parallel spin electrons having higher priority. For example, if there are three alpha and two beta electrons in an energy level, the three alpha electrons are distributed with respect to the angular probability distributions of the three sp^2 orbitals in the xz plane, while the two beta electrons are distributed with respect to those of the sp orbitals along the y axis.

Once the radial and angular coordinates for the electrons of an energy level have been assigned, they are converted to Cartesian coordinates. The entire energy level is then given a random rotation about a random axis. This rotation is easily computed using quaternions and prevents the distribution from becoming skewed along any axis or plane.

STRAW has been implemented in QMcBeaver, an open source program.[13] Researchers interested in further details of the algorithm are encouraged to download and examine the source code.

Equilibration Behavior

The computational experiments described in this section comparing the performance of the initialization methods were conducted using QMcBeaver.[13]

The VMC trial functions used in this section have the form $\Psi_{VMC} = \Psi_{SCF}J$, where Ψ_{SCF} is a HF/6-311G**++ wavefunction calculated using Jaguar[31] and J is a Pade-Jastrow correlation function with terms for each pair of particles in the molecule:

$$J = \exp\left(\sum_i \sum_{j<i} u_{ij}\right) \tag{12}$$

$$u_{ij} = \frac{c_{ij} r_{ij}}{1 + b_{ij} r_{ij}} \tag{13}$$

To satisfy the cusp condition[22] for an electron approaching a nucleus, we set $c = -Z$ for the electron–nuclear u functions, where
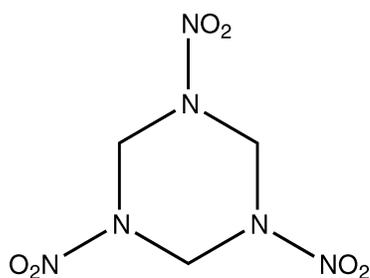
Figure 2. The RDX molecule.

Figure 3. Local energies of two RDX walkers. The walker initialized with GAWI starts off with very high local energy and approaches equilibration after several thousand steps, while the walker initialized with STRAW reaches a constant distribution very quickly.

Z is the charge of the nucleus. Similarly, we set $c = \frac{1}{2}$ for opposite spin electron pairs and $c = \frac{1}{4}$ for same spin electron pairs.

For opposite spin electron pairs, we use $b = 3.0$, and for same spin electron pairs and all nuclear–electron terms, we use $b = 100.0$. Our experience is that these values work reasonably well for ground states of molecules composed of atoms from the first three rows of the periodic table.

This is a very simple trial wavefunction, and its parameters are not optimized. To calculate accurate electronic properties for these molecules, the trial function could be improved by modifying the orbitals to satisfy the electron–nucleus cusp condition,[34] using a better Jastrow form,[19, 21] and optimizing its parameters.[16–18] In this work, however, we are focusing on equilibration and our ability to sample a wavefunction, so the simple trial function is sufficient.

In the calculations of this section, we use a time step of 0.001 au for both the equilibration and propagation phases, which results in propagation phase acceptance probabilities of 85% for $SiCl_4$ and 93% for RDX. Methods such as using a larger time step during the equilibration phase can be used to accelerate equilibration. To simplify comparisons between initialization methods, however, we use a constant time step in all of our calculations.

The effort that has gone into the more complicated initialization scheme pays off handsomely. Figures 3 and 4 show the behavior of the local energy of VMC walkers initialized with GAWI and STRAW. In each case, we find that walkers initialized with GAWI require several thousand steps to reach an equilibrium distribution, while walkers initialized with STRAW require very few.

Figure 3 shows the behavior of the local energy of two walkers during calculations on hexhydro-1,3,5-trinitro-1,3,5-triazine, or RDX (Fig. 2),[35] an energetic material. The local energy of the walker initialized with GAWI approaches a steady state after several thousand steps. This figure clearly shows the importance of avoiding contamination by the high energy nonequilibrated configurations in the beginning of the calculation. In contrast, the distribution of local energies for the walker initialized with STRAW is constant throughout the run. The initial configuration is in a region of high probability density and low local energy, and the long equilibration phase we see in the case of the GAWI walker is eliminated.

To test the effectiveness of STRAW on a molecule with atoms from the third row of the periodic table, a series of calculations was carried out with the $SiCl_4$ molecule. Figure 4 shows the behavior of the local energy of walkers initialized with GAWI and STRAW. Once again, we see that the local energy of the walker initialized with GAWI approaches equilibration after several thousand steps,
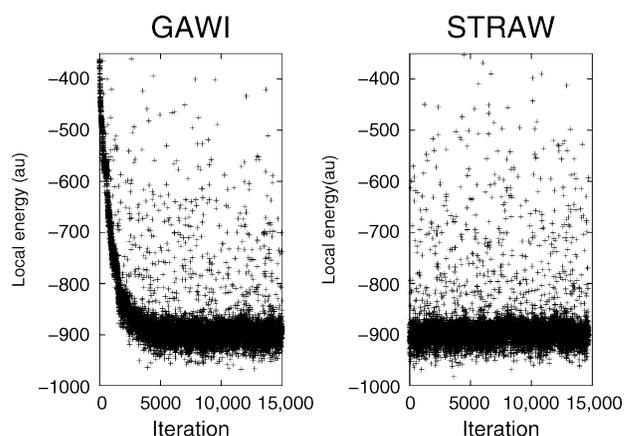
while the local energy of the walker initialized with STRAW reaches an equilibrium distribution very quickly.

Figures 3 and 4 examine only one walker for each initialization method. The results are encouraging, but a visual examination of the local energy is hardly a quantitative measure of equilibration. In addition, realistic QMC calculations on the molecules of this section will use ensembles of thousands to hundreds of thousands of walkers. To compare the behavior of ensembles of walkers generated by GAWI and STRAW, VMC calculations on the RDX molecule were carried out using the ASCI-QSC supercomputer at the Los Alamos National Laboratory. This machine is composed of 256 4 CPU HP/Compaq Alphaserver ES45's running at 1250 MHz. Calculations using 512 processors, 5 walkers per processor, and varying equilibration lengths were run until about 26 million samples were collected. The results are summarized in Table 1.
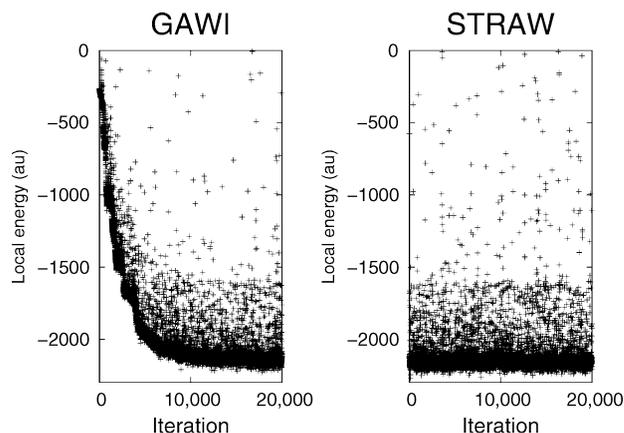

Figure 4. Local energies of $SiCl_4$ walkers. The local energy of the walker initialized with GAWI starts in a high energy region and approaches a steady state after several thousand steps, while the walker initialized with STRAW is equilibrated very quickly.

Table 1. VMC Calculations on RDX Were Carried Out Using 512 Processors with 5 Walkers per Processor to Compare Different Initialization Methods and Equilibration Lengths.

Initialization	Eq steps	$\langle E \rangle$ (au)	Total samples
GAWI	10 K	-893.114 ± 0.0122	26,324,421
GAWI	15 K	-893.235 ± 0.0103	26,334,855
GAWI	20 K	-893.275 ± 0.0102	26,024,948
GAWI	25 K	-893.291 ± 0.0184	26,272,857
GAWI	30 K	-893.296 ± 0.0101	26,291,124
STRAW	100	-893.298 ± 0.0099	25,145,777
STRAW	20 K	-893.291 ± 0.0117	26,071,024

Calculations with too few equilibration steps are contaminated by nonequilibrated samples and do not agree with calculations that are allowed to equilibrate. RDX calculations initialized with GAWI require 30,000 steps to equilibrate, while 100 steps are sufficient when STRAW is used.

Expectation values calculated using equilibrated walkers should be approximately independent of time, with random fluctuations. A long term, low frequency drift in an expectation value as samples are collected is a sign of contamination by nonequilibrated configurations. Figure 5 shows the energy expectation value versus the number of samples collected for these calculations. The left side of the figure shows the calculations initialized with GAWI. In the calculations with less than 30,000 equilibration steps, we see a monotonic decrease in the expectation value of the energy as points are collected. These calculations are contaminated with high energy, nonequilibrated configurations from the beginning of the calculation, and the expectation value decreases as equilibrated samples are added. The energy expectation value in the calculation with 30,000 equilibration steps has the desired behavior, fluctuating about the limit with no long term drift.

The calculations initialized with STRAW used 100 and 20,000 equilibration steps. We use a minimum of 100 equilibration steps in our calculations as a safety margin to be sure that each walker has at least one accepted move during the equilibration phase. The behavior of the energy expectation value in these calculations is very similar to that of the calculation initialized with GAWI using 30,000 equilibration steps. These three calculations show no signs of contamination. In Table 1, we see that their expectation values all agree to within one standard deviation of each other. These results demonstrate that 100 steps is sufficient for equilibration for RDX ensembles initialized with STRAW, while 30,000 equilibration steps are necessary when GAWI is used.

In QMcBeaver, standard deviations for expectation values are calculated using DDDA,[14] which averages samples into blocks in order to account for their serial correlation. If we examine the results for the calculations initialized with GAWI using 10,000, 15,000, and 20,000 equilibration steps in Table 1, we see that their energy expectation values do not agree with each other or those of the equilibrated calculations to within one standard deviation. This is important because it shows that the standard deviation calculated during a contaminated calculation does not necessarily reflect the inaccuracy of its expectation value. If a researcher specifies an equilibration phase that is too short and nonequilibrated configurations contaminate the statistics, the expectation values will be inaccurate,
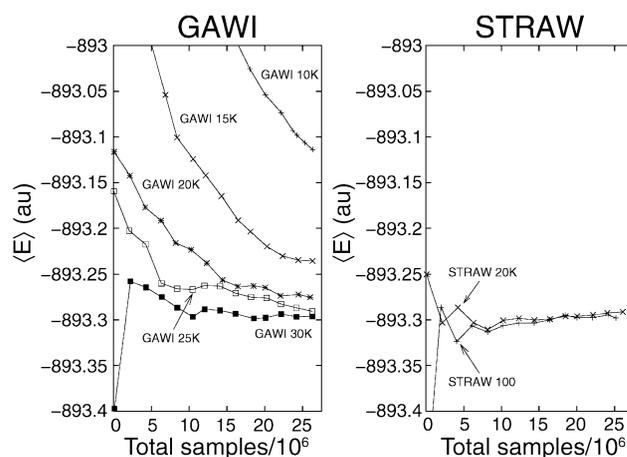

Figure 5. VMC calculations on RDX were carried out using 512 processors and 5 walkers per processor. Ensembles initialized with GAWI require 30,000 equilibration steps before contamination by high energy samples is eliminated. The ensemble initialized with STRAW is equilibrated after 100 steps.

and their standard deviations will not be a reliable measure of their inaccuracy.

The energy expectation value in the calculation that used GAWI and 25,000 equilibration steps agrees with the equilibrated results to within one standard deviation, but its behavior in Figure 5 still shows signs of contamination by high energy samples in the beginning of the run.

In contrast, Figure 6 replots the energy expectation value for two of the uncontaminated calculations. The error bars show the standard deviation of the expectation value. For these uncontaminated calculations, the eventual answer is within the range $\langle E \rangle \pm \sigma \langle E \rangle$
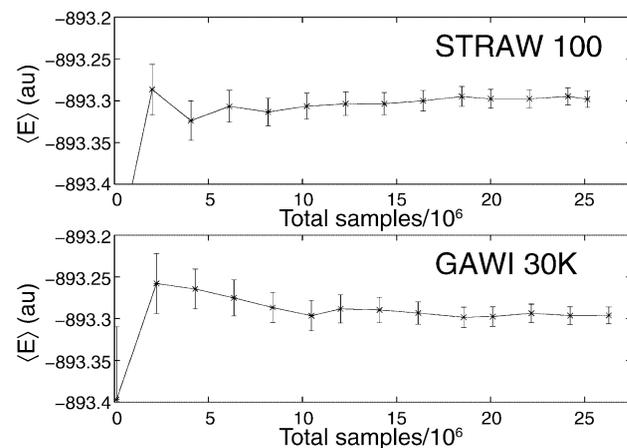
Figure 6. The energy expectation values and standard deviations for two uncontaminated RDX VMC calculations are shown. The calculations used 512 processors with 5 walkers per processor. The eventual answer is within one standard deviation of the expectation value at almost every point during the calculations. This shows that for uncontaminated calculations, the standard deviation of the energy expectation value is a good measure of its convergence.

at almost every point. In an uncontaminated calculation, we see that the standard deviation calculated by DDDA as the calculation progresses is a good measure of the level of convergence of the expectation value.

Timing and Spatial Correlation

Because it is more complicated than GAWI, STRAW takes more time to generate an initial electronic configuration for a walker. The new initialization method would be of little use if the time it took to generate an initial configuration was greater than the time saved in equilibration steps. Although coordinates are generated for each electron individually, the use of splines makes the process very inexpensive. Generating an initial configuration using STRAW requires less time than two VMC iterations for each of the molecules examined in this work.

The equilibration phase of a QMC calculation allows the walkers to become independent of their initial configurations and, by extension, each other. Since our objective is to shorten the equilibration phase of the calculation, an important objection to using STRAW could be raised if it led to spatial correlation, or any kind of statistical dependence within the ensemble of walkers.

Testing for spatial correlation in an ensemble of walkers is difficult. Vectors can be tested for spatial correlation by taking dot products, but a comparison of electronic configurations must take into account the indistinguishability of identical particles and the symmetry of the molecule. We avoid these difficulties by instead examining the statistical consequences of spatial correlation. If the walkers are indeed independent of each other, we expect that for a fixed number of iterations, the standard deviation of the energy expectation value will be proportional to $\frac{1}{\sqrt{W}}$, where W is the number of walkers. Any spatial correlation among the walkers will result in a different trend.

Single processor VMC calculations with 200,000 propagation steps and different ensemble sizes were carried out using the ethanol molecule. The trial function has the form $\Psi_{\text{VMC}} = \Psi_{\text{SCF}}J$, where Ψ_{SCF} is a HF/6-311G**++ wavefunction calculated using Jaguar[31] and J is the Jastrow function described in eqs. (12) and (13). One series of calculations was initialized with GAWI and allowed to equilibrate for 200,000 steps. We assume the walkers in these ensembles are independent of each other and their initial configurations. A second series was started with ensembles of identical walkers. These ensembles start with perfect spatial correlation. A third series of calculations was initialized with STRAW and used 100 equilibration steps.

Figure 7 shows the results for the different ensembles. The points for the equilibrated and STRAW ensembles are close to each other and the function $0.20 \times W^{-0.5}$, which is what we expect for independent walkers. The points for the identical ensembles, on the other hand, are very close to the function $0.24 \times W^{-0.3}$. Because they do not sample as much configuration space as independent walkers, ensembles with a high degree of spatial correlation generate less information than ensembles that are independent. Although the equilibration phase is very short, the statistical behavior of the STRAW ensembles is very similar to that of the equilibrated ensembles and to the behavior expected of independent walkers.

The initial electronic configurations generated by STRAW are statistically independent of each other and in regions of high enough
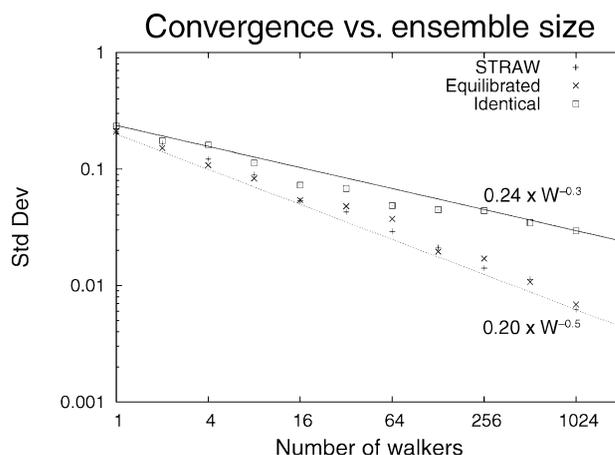
Figure 7. Standard deviation of energy expectation values for single processor ethanol calculations using equilibrated, identical, and STRAW ensembles after 200,000 propagation steps. The points for the equilibrated and STRAW ensembles are very close to the function $0.20 \times W^{-0.5}$, which shows that walkers generated by STRAW are statistically independent of each other.

probability density that a long equilibration phase is not necessary. The initialization algorithm is based on general principles of electronic structure, such as energy levels and the Pauli principle. It does not, however, generate configurations directly with respect to ρ_{VMC}, and is not meant to substitute for Metropolis sampling.

Parallel Calculation Efficiency

The equilibration phase of a QMC calculation is an inherently serial step—the walkers on each processor must be equilibrated individually, so adding more processors increases the time spent on this phase of the calculation. Knowing the appropriate number of iterations to exclude is vital, because leaving out too many wastes computer time, while leaving out too few will result in nonequilibrated values contaminating the statistics.

The calculations of Sections "Equilibration Behavior" and "Timing and spatial correlation" give us confidence that STRAW can generate independent initial configurations for RDX and SiCl₄ in regions of high enough density that one hundred equilibration steps is sufficient before calculating expectation values. We expect that the electronic structure of other molecules composed of atoms from the first three rows of the periodic table will be similar enough to these examples to allow STRAW to be successful for them as well.

An automatic method for determining the equilibration length based on the statistical behavior of the ensemble of walkers is being developed. This will remove the need for the user to specify the number of equilibration steps and ensure the highest possible accuracy and efficiency in parallel calculations.[36]

To demonstrate the effect of shortening the equilibration phase of a calculation, a scaling experiment was performed on ASCI-QSC. VMC calculations on RDX were conducted using 1,000,000 total propagation steps and 1 walker per processor. Following the results of Section "Equilibration Behavior", 30,000 equilibration steps were used in the calculations initialized with GAWI, while

Efficiency of parallel RDX calculations
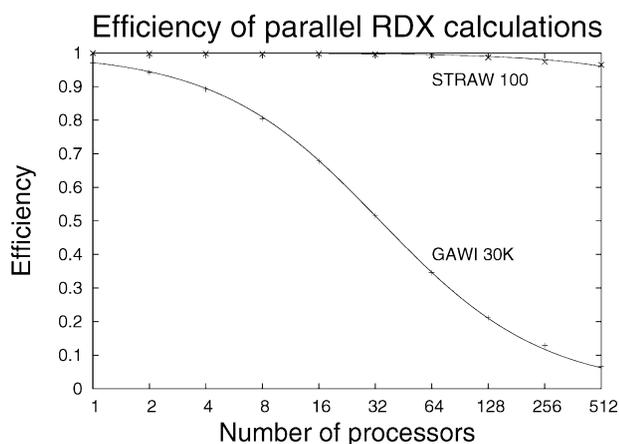
Figure 8. RDX calculations with one walker per processor were run until 1,000,000 total samples were collected. The calculations initialized with GAWI used 30,000 equilibration steps, while the calculations initialized with STRAW used 100 equilibration steps. Decreasing the number of equilibration steps greatly improves the efficiency of calculations with large numbers of processors. The data are fit to $\epsilon(N_{\text{Processors}}) = \frac{a}{a + N_{\text{Processors}}}$ with $a = 34.0$ for GAWI and $a = 12,514$ for STRAW.

100 equilibration steps were used with STRAW. The efficiency of each calculation was found using eq. (7).

The points in Figure 8 were fit to the function $a/(a + N_{\text{Processors}})$. The value for a for GAWI is 34.0, while for STRAW it is 12,514. This result clearly shows the effect of reducing the number of equilibration steps on the efficiency of parallel calculations. The experiment has a short statistics gathering phase, which makes it scale particularly badly as the number of processors increases. In a realistic calculation on RDX, many more steps will have to be used before the expectation values converge to within chemical accuracy. A calculation with a longer statistics gathering phase will scale more favorably as the number of processors increases, which can be seen by examining eq. (7). As computers with large numbers of processors come into general use,[11, 12] however, the equilibration phase will limit the efficiency of any calculation.

The most important consequence of reducing the length of the equilibration phase with STRAW is that the improvement in parallel efficiency will speed the calculation of converged expectation values. Using an automatic method to terminate the calculation based on the convergence of the energy expectation value,[14, 15] RDX calculations using 512 processors with five walkers per processor were run until the expectation value of the energy converged to within

0.01 au or 6.27 kcal/mol. 30,000 equilibration steps were used in the calculations initialized with GAWI, while 100 equilibration steps were used with STRAW.

Table 2 summarizes the results from these calculations, and Figure 9 shows the standard deviation of the energy expectation value versus total iterations on the root processor. The total iterations include the equilibration phase, and we see that the two calculations have very similar convergence behavior, with the calculation initialized with GAWI offset by about 30,000 iterations compared to the one initialized with STRAW. The calculation initialized with STRAW converged to the desired level in 6.3 h with 99.8% efficiency, while the calculation initialized with GAWI took 9.4 h with 65.0% efficiency.

Conclusion

We have presented and tested STRAW, a simple and automatic method for generating initial electronic configurations for QMC calculations. STRAW is based on the structure of the energy levels of atoms and distributes the electrons in annular shells. The electrons in each energy level are distributed with respect to probability distributions in the angular coordinates so that they avoid each other. The configurations generated by STRAW are statistically independent of each other and are in regions of high probability density, which reduces the length of the equilibration phase of the calculation, during which the statistics must be discarded. STRAW has been implemented in QMcBeaver, an open source QMC program.[13]

Using an appropriate equilibration length is vital, because when the statistics are contaminated by nonequilibrated configurations, both the expectation values and their standard deviations can be inaccurate. STRAW simplifies the job of the user to specify the equilibration length by generating initial configurations that show no signs of contamination or spatial correlation after an equilibration phase of one hundred iterations.
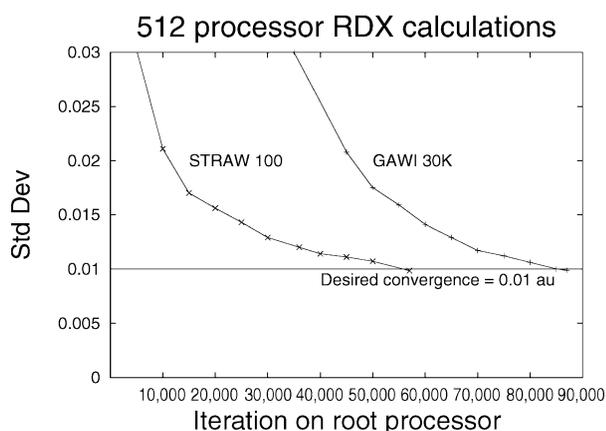
Shortening the equilibration phase increases the efficiency of parallel QMC calculations and decreases the amount of computer time needed to calculate converged expectation values. For example, using STRAW instead of a standard initialization method in 512 processor calculations decreases the time needed to calculate the energy expectation value of a trial function for an RDX molecule to within 0.01 au from 9.4 h to 6.3 h, an improvement of 33%.

Using STRAW improves the parallel scaling of QMC and will increase the efficiency of calculations using tens to hundreds of thousands of processors. This will in turn allow highly accurate simulations on a broader range of chemically interesting systems than is possible today. QMC results will be useful as benchmarks

Table 2. RDX Calculations Using 512 Processors and 5 Walkers per Processor Were Run Until 0.01 au Convergence in the Energy Expectation Value was Achieved.

Initialization	Eq steps	$\langle E \rangle$ (au)	Steps on root proc	Eff (%)	Wall clock time
GAWI	30 K	-893.291 ± 0.0099	85,500	65.0	9.40 h
STRAW	100	-893.287 ± 0.0098	57,048	99.8	6.30h

The calculation initialized with GAWI used 30,000 equilibration steps, while the calculation initialized with STRAW used 100 equilibration steps. The calculation initialized with STRAW took 6.3 h to converge, while the calculation initialized with GAWI took 9.4 h.

Figure 9. VMC calculations on RDX were carried out using 512 processors and 5 walkers per processor. The total iterations on the *x* axis include the equilibration phase of the calculations. Initializing the walkers with STRAW decreases the wall clock time needed to calculate the RDX energy expectation value to within 0.01 au from 9.4 to 6.3 h, an improvement of 33%.

for training force fields for molecular dynamics simulations and developing new density functional (DFT) methods. There are several classes of systems which have proven elusive for current DFT methods.[37,38] Reproducing QMC results for these systems will be an important goal for the next generation of DFT methods.

Clearly, many other schemes for generating initial electronic configurations for QMC calculations are possible. The tests for equilibration of the Markov chain, initialization time, and spatial correlation described in the Sections "Equilibration Behavior" and "Training and Spatial Correlation" will provide a basis for comparison of future initialization schemes.

References

1. Anderson, J. B. J Chem Phys 1976, 65, 4121.
2. Ceperley, D. M.; Alder, B. J. Phys Rev Lett 1980, 45, 566.
3. Grossman, J. C. J Chem Phys 2002, 117, 1434.
4. Lu, Shih-I. J Chem Phys 2002, 118, 9528.
5. Manten, S.; Lüchow, A. J Chem Phys 2001, 115, 5362.
6. Manten, S.; Lüchow, A. J Chem Phys 2003, 119, 1307.
7. Williamson, A. J.; Hood, R. Q.; Grossman, J. C. Phys Rev Lett 2001, 87, 246406.
8. Reboredo, F. A.; Williamson, A. J. Phys Rev B 2005, 71, 121105.
9. Aspuru-Guzik, A.; Salomon-Ferrer, R.; Austin, B.; Lester, W.A., Jr. J Comput Chem 2005, 26, 708.
10. Morokuma, K.; Musaev, D. G.; Vreven, T.; Basch, H.; Torrent, M.; Khoroshun, D. V. IBM J Res Dev 2001, 45, 367.
11. Teragrid. 2008. 8 April 2008 http://www.teragrid.org.
12. Blue Gene. 6 December 2007. IBM Research. 8 April 2008 http://www.research.ibm.com/bluegene.
13. Feldmann, M. T.; Kent, IV, D. R.; Fisher, D. R.; Anderson, A. G. QM^{c}Beaver v20051107 ©, 2001–2007 http://sourceforge.net/projects/qmcbeaver.
14. Kent, D. R., IV.; Muller, R. P.; Anderson, A. G.; Goddard, W. A., III.; Feldmann, M. T. J Comput Chem 2007, 28, 2309.
15. Feldmann, M. T.; Cummings, J. C.; Kent, D. R., IV.; Muller, R. P.; Goddard, W. A., III. J Comput Chem 2008, 29, 8.
16. Umrigar, C. J.; Wilson, K. G.; Wilkins, J. W. Phys Rev Lett 1988, 60, 1719.
17. Umrigar, C. J.; Filippi, C. Phys Rev Lett 2004, 94, 150201.
18. Toulouse, J.; Umrigar, C. J. J Chem Phys 2007, 126, 084102.
19. Myers, C.R.; Umrigar, C. J.; Sethna, J. P.; Morgan, J. D., III. Phys Rev A 1991, 44, 5537.
20. Huang, C.-J.; Umrigar, C. J.; Nightingale, M. P. J Chem Phys 1997, 107, 3007.
21. Drummond, N. D.; Towler, M. D.; Needs, R. J. Phys Rev B 2004, 70, 235119.
22. Pack, R. T.; Brown, W. B. J Chem Phys 1966, 45, 556.
23. Reynolds, P. J.; Ceperley, D. M.; Alder, B. J.; Lester, W. A., Jr. J Chem Phys 1982, 77, 5593.
24. Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M. N.; Teller, A. H.; Teller, E. J Chem Phys 1953, 21, 1087.
25. Umrigar, C. J. Phys Rev Lett 1993, 71, 408.
26. Umrigar, C. J.; Nightengale, M. P.; Runge, K. J. J Chem Phys 1993, 99, 2865.
27. Needs, R.; Rajagopal, G.; Towler, M. D.; Kent, P. R. C.; Williamson, A. CASINO, the Cambridge Quantum Monte Carlo Code, version 1.1.0, 2000.
28. Barnett, R. N.; Hammond, B. L.; Reynold, P. J.; Terray, L.; Lester, W. A., Jr. QMAGIC, the Quantum Monte Carlo code, version 1.1.0, 2000.
29. Aspuru-Guzik, A.; Salomon-Ferrer, R.; Austin, B.; Perusquia-Flores, R.; Griffin, M.A.; Oliva, R.A.; Skinner, D.; Domin, D.; Lester, W.A., Jr. J Comput Chem 2005, 26, 856.
30. Press, W.H.; Teukolsky, S.A.; Vetterling, W.T.; Flannery, B.P. Numerical Recipes in C: The Art of Scientific Computing; Cambridge University Press; Cambridge, UK, 2002.
31. Ringnalda, M. N.; Langlois, J.-M.; Murphy, R. B.; Greeley, B. H.; Cortis, C.; Russo, T.V.; Marten, B.; Donnelly, R. E., Jr.; Pollard, W.T.; Cao, Y.; Muller, R.P.; Mainz, J.R.; Wright, J. R.; Miller, G.H.; Goddard, W. A., III; Friesner, R.A. Jaguar v4.1, 2001.
32. Boys, S. F. Rev Mod Phys 1960, 32, 296.
33. Linus, P. J Am Chem Soc 1931, 53, 1367.
34. Ma, A.; Towler, M. D.; Drummond, N. D.; Needs, R. J. J Chem Phys 2005, 122, 224322.
35. Strachan, A.; Kober, E. M.; van Duin, A. C. T.; Oxgaard, J.; Goddard, W. A., III. J Chem Phys 2005, 122, 054502.
36. Fisher, D.R.; Kent, D.R., IV; Feldmann, M.T.; Goddard, W.A., III. Manuscript in Preparation.
37. King, R. A.; Crawford, T. D.; Stanton, J. F.; Schaefer, H. F., III. J Am Chem Soc 1999, 121, 10788.
38. Černý, J.; Hobza, P. Phys Chem Chem Phys 2005, 7, 1624.